Data Analysis Report

dataMineR

4/6/2013

CONTENTS

Contents **2**

1 Data Analysis Report **3**
1.1 Introduction . . . 3
1.2 Dataset Basic Artifacts . . . 3
1.2.1 Basic dataset information . . . 3
1.3 Variabele types . . . 4
1.4 Excluded variables . . . 5

2 Numeric variables **6**
2.1 Overview . . . 6
2.2 Variabele age . . . 7
2.3 Variabele income . . . 8
2.4 Variabele y . . . 9

CHAPTER 1

DATA ANALYSIS REPORT

This is the first attempt to transform the Rmd source for dataMineR into a pdf file. Some things are not yet as they should be: html tables are obviously not converted properly.

Other things work out of the box: syntax highlighting (although not yet accurate) and TOC generation.

1.1 Introduction

This data analysis report is generated using R-studio and knitr to knit R code and markdown into html format. We have the option to include all R code that is used to generate the plots and calculations. Default this feauture is dissabled. The data analysis step is the first step an a datamining analysis.

1.2 Dataset Basic Artifacts

Basic information from the dataset we are using.

1.2.1 Basic dataset information

```
i = 1

# # data location full path to filename from working directory
    (=project
# dir)
path2file <- "../data/data-simple-example.tab"

# read dataframe from tab delimets file
data <- read.delim(path2file, sep = "\t", strip.white = TRUE)

# determine number of rows and colums in dataframe
rows <- nrow(data)
```

```
colums <- ncol(data)

# case_id = registrnr
original_case_id = "caseID"
# data$caseID <- data$caseID

# check if case_id is unique
if (!(length(unique(data$caseID)) == length(data$caseID))) {
    cat("Warning : Case_id appears not unique ! ")
}

# exclude original case_id and variables with lot of missing
# exclude_var_names <-
# c('caseID','registrnr','X2011tmoktstornaant','
   X2010stornoaantal')
exclude_var_names <- c("caseID", "p_y", "p_real")
data <- data[, !names(data) %in% exclude_var_names]
```

We are using data from file : ../data/data-simple-example.tab The dataset has 7 variables and 5000 rows.

The case identifyer is *caseID* this is unique for all cases.

1.3 Variabele types

```
# names in header
var_names <- names(data)

# sometimes variabels are in the dataset as codes, they appear
    numeric but
# code for a category

## treat_as_categorical <-
## c('catHHINKOMEN','catHHSOCIALE','catHHOPLEIDI','
   catHHLEVENSF','catHHGEOTYPE','catHHTYPEWO',
##'catHHEIGENDO','catHHWOZWAA', 'catBELEGGERS','catLENERS',
   'catSPAARDERS','catSWITCHGEVO'
## ,'catMERKENTROU')
treat_as_categorical <- NULL

# transform numeric into factors
data[treat_as_categorical] <- lapply(data[treat_as_categorical
   ], as.factor)

num_var_names <- names(data[sapply(data, is.numeric)])
num_vars <- length(num_var_names)
cat_var_names <- names(data[sapply(data, is.factor)])
cat_vars <- length(cat_var_names)
```

The following variabeles are present in the dataset: age, income, gender, y

We have 3 numeric variables and 1 categorical variables (or factors in R).

1.4 Excluded variables

From the varables provided the folowing list will be excluded in this anlysis: caseID, p_y, p_real

Sometimes categoric variables are present as coded numbers. These should be treated as factors. In this dataset the following variables will be used as factors(categoric):

We have 3 numeric variables and 1 categorical variables (or factors in R).

CHAPTER 2

NUMERIC VARIABLES

Here we analyse all numeric variables. We start with an overview on basic statistics per variable. We check for missing values. We do a histogram plot to show the distribution for this variable. And we test for outliers.

2.1 Overview

In the table below we report the number of observations (n), the smallest observation (min), the first quantile (q1), the media , the mean, last quantile, the largest observation (max), and the nber of missing values (na).

```
library(xtable)

# summarize numeric variables
td <- data[, sapply(data, is.numeric)]
td.min <- sapply(td, min, na.rm = TRUE)
td.mean <- sapply(td, mean, na.rm = TRUE)
td.median <- sapply(td, median, na.rm = TRUE)
td.max <- sapply(td, max, na.rm = TRUE)
td.n <- as.numeric(apply(td, 2, function(x) length(which(!is.
    na(x)))))
td.na <- as.numeric(apply(td, 2, function(x) length(which(is.
    na(x)))))
td.q <- apply(td, 2, quantile, na.rm = TRUE)

tddf <- data.frame(cbind(td.n, td.na, td.min, td.mean, td.
    median, td.max))
names(tddf) <- c("n obs", "n missing", "min", "mean", "median"
    , "max")

print(xtable(tddf), type = "html")
```

n obs n missing min mean median max age 5000.00 0.00 18.01 42.81 40.78 84.98 income 5000.00 0.00 14556.59 34062.62 32953.06 83610.97 y 5000.00 0.00 0.00 0.97 1.00 1.00

```
## run numeric template for each numeric variable seperately
```

2.2 Variabele age

Missing : 0
Minimum value : 18.0131
Percentile 1 : 18.5014
Percentile 99 : 81.9667
Maximum value : 84.9796

```
warn_extreme_values = 3
d1 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[2] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[1]
d99 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[101] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[100]
if (d1) {
    cat("Warning : Suspect extreme values in left tail")
}
if (d99) {
    cat("Warning : Suspect extreme values in right tail")
}
```

```
library(ggplot2)
```

```
## Loading required package: methods
```

```
v <- num_var_names[i]
hp <- ggplot(na.omit(data), aes_string(x = v)) + geom_
    histogram(colour = "grey",
    fill = "grey", binwidth = diff(range(na.omit(data[[v]]))/
        100))

hp + theme(axis.title.x = element_blank(), axis.text.x =
    element_text(size = 10)) +
    theme(axis.title.y = element_blank(), axis.text.y =
        element_text(size = 10))
```

```
## Warning: position_stack requires constant width: output may
     be incorrect
```

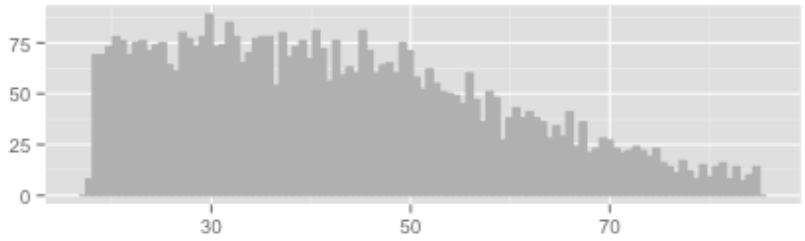

Figure 2.1: plot of chunk unnamed-chunk-2

2.3 Variabele income

Missing : 0
Minimum value : 1.4557 × 104
Percentile 1 : 1.8644 × 104
Percentile 99 : 5.9851 × 104
Maximum value : 8.3611 × 104

```
warn_extreme_values = 3
d1 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[2] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[1]
d99 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[101] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[100]
if (d1) {
    cat("Warning : Suspect extreme values in left tail")
}
if (d99) {
    cat("Warning : Suspect extreme values in right tail")
}
```

```
library(ggplot2)

v <- num_var_names[i]
hp <- ggplot(na.omit(data), aes_string(x = v)) + geom_
    histogram(colour = "grey",
    fill = "grey", binwidth = diff(range(na.omit(data[[v]]))/
        100))

hp + theme(axis.title.x = element_blank(), axis.text.x =
    element_text(size = 10)) +
    theme(axis.title.y = element_blank(), axis.text.y =
        element_text(size = 10))
```

```
## Warning: position_stack requires constant width: output may
     be incorrect
```

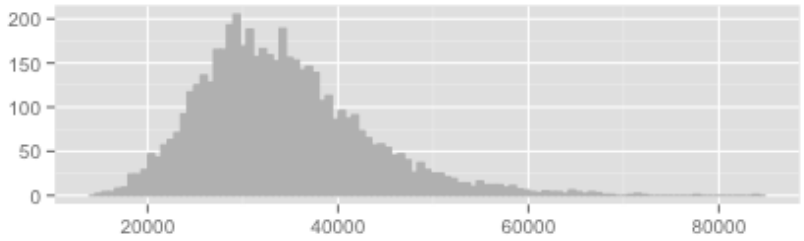

Figure 2.2: plot of chunk unnamed-chunk-4

2.4 Variabele y

Missing : 0
Minimum value : 0
Percentile 1 : 0
Percentile 99 : 1
Maximum value : 1

```
warn_extreme_values = 3
d1 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[2] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[1]
d99 = quantile(na.omit(data[[num_var_names[i]]]), probs = seq
    (0, 1, 0.01))[101] >
    warn_extreme_values * quantile(na.omit(data[[num_var_names
        [i]]]), probs = seq(0,
        1, 0.01))[100]
if (d1) {
    cat("Warning : Suspect extreme values in left tail")
}
if (d99) {
    cat("Warning : Suspect extreme values in right tail")
}
```

```
library(ggplot2)

v <- num_var_names[i]
hp <- ggplot(na.omit(data), aes_string(x = v)) + geom_
    histogram(colour = "grey",
    fill = "grey", binwidth = diff(range(na.omit(data[[v]]))/
        100))

hp + theme(axis.title.x = element_blank(), axis.text.x =
    element_text(size = 10)) +
    theme(axis.title.y = element_blank(), axis.text.y =
        element_text(size = 10))
```

```
## Warning: position_stack requires constant width: output may
     be incorrect
```

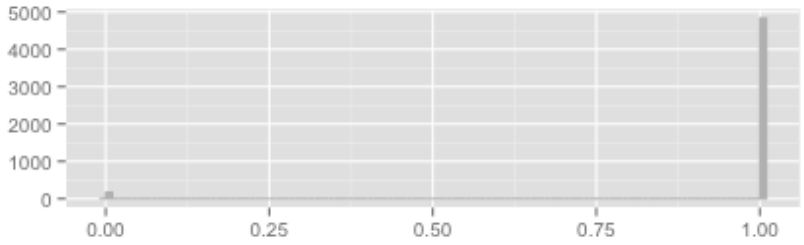

Figure 2.3: plot of chunk unnamed-chunk-6